Introduction to "Notes on the Design of Optimal FIR Filters"

Introduction

A recurring technical task in the development of digital signal processing products and systems is the design of finite-impulse-response (FIR) digital filters. Fortunately some excellent software packages exist for the automatic synthesis of impulse responses for such filters, many of them based on the now-famous Parks-McClellan algorithm [2]. Unfortunately, there is still some mystery about how to use the software and, equally important, how to estimate impulse response lengths short of actually designing the filter itself. This technical note primarily addresses the second problem and indirectly discusses a bit the first. We examine here how to convert a typical filter specification in terms of cutoff frequency, passband ripple, etc., into a reasonably accurate estimate of the length of the impulse response. Not only does this estimate suffice for most design tradeoff exercises, it usually allows the Parks-McClellan routines to be employed only once or twice rather than the multiple times needed when the “cut-and-try" method is used.

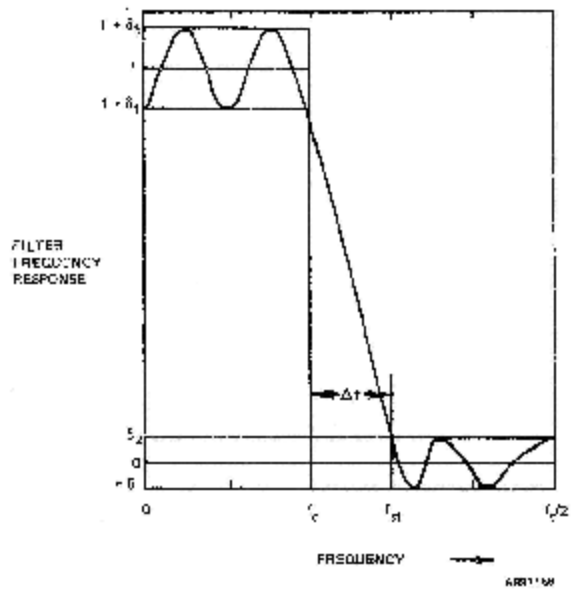
Statement of the Optimal Linear Phase FIR Filter Design Problem

Equal-ripple Design

While other types of filters are often of interest, this note focuses on the lowpass linear phase filter. Even though it is not immediately obvious, virtually all of the analytical results developed in this note apply to the other types as well. This fact is amplified in the module Extension to Non-lowpass Filters.

It is known that the Parks-McClellan filter synthesis software package produces “optimal" filters in the sense that the best possible filter performance is attained for the number of “filter taps" allowed by the designer. “Optimal" can be defined various ways. The Parks-McClellan package uses the Remez exchange algorithm to optimize the filter design by selecting the impulse response of given length, termed here N, which minimizes the peak ripple in the passband and stopband. It can be shown, though not here, that minimizing the peak, or maximum, ripple is equivalent to making all of the local peaks in the ripple equal to each other. This fact leads to three different names for essentially the same filter design. They are commonly called “equal-ripple" filters, because the local peaks are equal in deviation from the desired filter response. Because the maximum ripple deviation is minimized in this optimization procedure, they are also termed “minimax" filters. Finally, since the Russian Chebyshev is usually associated with minimax designs [footnote], these filters are often given his name.
He developed the concept of minimax design and a set of polynomials which carry his name not from filter design, but from the optimal design of piston drive rods for steam locomotives. They are discussed more in "Filter Sizing" and Appendix B.

The design template for an equal-ripple lowpass filter is shown in [link].



Frequency Response of an
Optimal Weighted Equal-ripple
Linear Phase FIR Filter

The passband extends from 0 Hz to the cutoff frequency denoted f_c. The gain in the passband is assumed to be unity. Any other gain is attained by scaling the whole impulse response appropriately. The stopband begins at the frequency denoted f_{st} and ends at the so-called Nyquist or “folding" frequency, denoted by $\frac{f_s}{2}$, where f_s is the sampling frequency of the data entering the digital filter. In some references, [1] for example, the sampling rate f_s is assumed to be normalized to unity just as the passband gain has here. The dependence on the sampling frequency is kept explicit in this note, however, so that its impact on design parameters can be kept visible.

The optimal synthesis algorithm is assumed here to produce an impulse response whose associated frequency response has ripples in both the passband and the stopband. The peak deviation in the passband is denoted δ_1 and the peak deviation in the stopband is denoted δ_2. It is commonly thought that an “equal-ripple" design forces δ_1 to equal δ_2. In fact this is not true. The local ripple peaks in the passband will all equal δ_1 and those in the stopband will all equal δ_2. For a given filter specification the two are linked

together by a weight denoted W, so that $\delta_1 = W\delta_2$. In fact the Parks-McClellan routines insure the design of **weighted equal-ripple** filters. The choice of W is discussed shortly.

An important design parameter is the **transition band**, denoted Δf, and defined as the difference between the stopband edge f_{st} and the passband edge f_c. Thus,
Equation:

$$\Delta f = f_{st} \ - \ f_c.$$

In theory the required filter order N is a function of all of the design parameters defined so far, that is, f_s, f_c, f_{st}, δ_1, and δ_2. The central point of this technical note is that under a large range of practical circumstances the required value of N can be estimated using only f_s, Δf, and the smaller of δ_1 and δ_2.

Conversion of Specifications

While the parameters defined in the previous section relate directly to the theory of FIR filter design optimization, some of them differ from those usually employed to specify the performance of a filter. We discuss here the conversion of two of those, δ_1 and δ_2, into more traditional measures.

Passband Ripple: [link] uses the parameter δ_1 to describe the peak difference between the template lowpass filter and the magnitude of the filter response actually attained. Traditionally this passband ripple has been specified in terms of the maximum difference in the power level transmitted through the filter in the passband. By this definition, the peak-to-peak passband ripple, abbreviated here as PBR, is given by
Equation:

$$PBR = 10 \ log_{10} \frac{(1+\delta_1)^2}{(1-\delta_1)^2}.$$

Assuming that the nominal power transmission through the filter is unity, the numerator is the power gain at a ripple peak and the denominator is the gain at a trough. It is easily shown (see Appendix A) that when δ_1 is small compared to unity, or, equivalently, when the passband ripple is less than about 1.5 dB, then [footnote]
Strictly speaking, the peak ripple excursions are equal in magnitude, not in decibels. This subtlety is completely negligible for small values of δ_1.
Equation:

$$PBR \approx 17.36\ \delta_1.$$

Stopband Ripple: The traditional specification for stopband ripple, abbreviated here as SBR, is the power difference between the nominal passband transmission level and the transmission level of the highest ripple in the stopband. For the equal ripple design shown in [link], all stopband ripples have equal peak values and the nominal passband transmission is unity, that is, 0 dB. The stopband ripple, or more accurately, the minimum stopband power rejection, denoted SBR, is given by
Equation:

$$SBR = 20\ log_{10}\delta_2.$$

Example:
Suppose a filter is specified to have a peak-to-peak passband ripple of 0.5 dB and a minimum stopband attenuation of 60 dB. Using the above equations we find that $\delta_1 = 0.0288$, $\delta_2 = .001$, and the relative weighting, W, therefore equals 28.8. ♠

In discussing filter specifications it should be noted that the cutoff frequency f_c shown in [link] differs from the definition typically used in analog filter designs. The cutoff frequency is commonly defined as the **3 dB point**, that is, that frequency at which the power transfer function falls to a

value 3 dB below the nominal passband level. Instead the value of f_c shown in [link] is the highest frequency at which the specified passband ripple is still attained. In very few practical cases do the two definitions result in the same value.

Filter Sizing

The Formula for Estimation of the FIR Filter Length

For many lowpass filter designs the peak passband excursion δ_1 exceeds the peak stopband excursion δ_2 by a factor of ten or more. This ratio, earlier denoted as the weight W, was just evaluted in the previous section to have the value 28.8 for a typical set of specifications. In this example the stopband attenuation specification drives the required filter order. In this case, and with a few additional assumptions which will be enumerated later, the number of coefficients in the impulse response of a high-order FIR linear phase filter, denoted N, can be accurately estimated using the formula:
Equation:

$$N \approx \frac{\alpha f_s}{\Delta f},$$

where the design parameter α is given by the equation:
Equation:

$$\alpha = 0.22 + 0.0366 \cdot SBR.$$

As before, SBR is the minimum stopband attenuation compared to the nominal passband power transmission level, measured in decibels.

Example:
Continuing from Example I "Statement of the Optimal Linear FIR Filter Design Problem"
Suppose as before that the lowpass filter of interest is to have a peak-to-peak passband ripple (PBR) of 0.5 dB and a minimum stopband attenuation of 60 dB. Since W has been evaluated to be approximately 29 in this case, [link] applies. Using [link], α is evaluated to be 2.42. Thus N is closely approximated by 2.42 times the reciprocal of the normalized transition bandwidth $\frac{\Delta f}{f_s}$. To continue the example assume that the sampling rate is 8 kHz, that the cutoff frequency f_c is 1530 Hz, and that the stopband edge f_{st} is 2330 Hz. Thus $\Delta f = 800$ Hz and $\frac{\Delta f}{f_s} = 0.1$, yielding an estimated filter order N of approximately 24. Executing the Parks-McClellan design program with these parameters happens to produce an impulse response which almost perfectly matches the desired result (e.g., peak stopband ripple of 60.07 dB as opposed to the stated objective of 60 dB).
♠

Note that the required filter order N as estimated by [link] and [link] does not depend on the passband ripple PBR or on the exact values of the cutoff and stopband frequencies. Thus, when the conditions allowing the underlying assumptions to be met are true, estimating the required filter order N becomes very easy.

[link] provides the values of the design parameter α from [link] for various degrees of stopband suppression. Given also is the range of the passband ripple for which the values of α apply. The

column marked **maximum passband ripple** reflects the the assumption that the passband deviation δ_1 is small compared to unity; specifically, the stated value of 1.74 dB corresponds to $\delta_1 = 0.1$. The rightmost column, denoted **minimum passband ripple**, is the limit imposed by the assumption that $\delta_1 > 10 \cdot \delta_2$. Of course FIR linear phase equal ripple filters can be designed with passband ripple extending beyond the stated range. However, as the PBR specification approaches either of these endpoints the validity of [link] will degrade. The predicted filter length will err on the low side for small PBR values and be overly pessimistic for PBR $>$ 1.74 dB. In such cases, an iteration on design might be necessary to obtain the desired filter characteristics.

Stopband Attenuation (in dB)	α	Maximum Passband Ripple (in dB)	Minimum Passband Ripple (in dB)
45	1.87	1.74	1.0
50	2.05	1.74	0.55
55	2.23	1.74	0.31
60	2.42	1.74	0.174
65	2.60	1.74	0.098
70	2.78	1.74	0.055

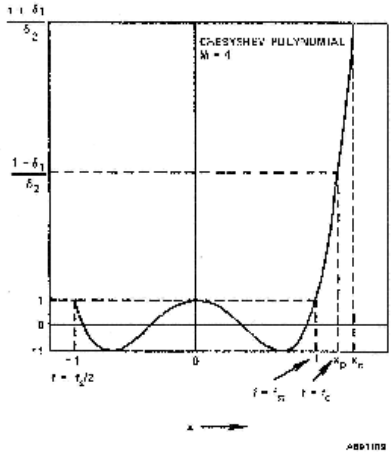
Table 1: Values of the Design Parameter α as a Function of the Minimum Stopband Attenuation

Derivation of the Formula

This section describes the theoretical underpinnings of [link] and [link]. A clear understanding of this section is not required to use the Parks-McClellan software routines or to enjoy the remainder of this technical note.

As discussed in Section 2, the Parks-McClellan synthesis algorithm uses the Remez exchange algorithm to optimally select the values of the N impulse response coefficients in such a way as to minimize the weighted peak difference between the desired magnitude frequency response and the actual one. Since the solution to this optimization problem does not have a closed form, it is not easy to generalize its properties. To learn about its properties and to develop appropriate design rules, McClellan, Rabiner, and others synthesized thousands of filters and measured their properties. Curves with this sort of information are presented in [1], along with a complicated empirical formula for the filter order N in terms of all of the parameters specifying the filter. While this work is not immediately useful for design work, a limiting case uncovered by those workers does provide some insight into the optimal filter solutions and leads to the simple rules compressed into [link] and [link].

Suppose we desire to design a high-order, FIR, linear phase filter for which the passband is as narrow as possible. Looking again at Figure 1 from the module titled "Statement of the Optimal Linear Phase FIR Filter Design Problem" with this in mind reveals that all of the ripple behavior for such a filter will occur in the stopband. Such a filter, or a very close approximation to it, can be synthesized using another FIR filter design method, that of multiplying a sampled $\frac{sin\ q}{q}$ function, where $q = \frac{\pi f}{f_s}$, by an N-point window function constructed from a Chebyshev polynomial. The sampled $\frac{sin\ q}{q}$, or sinc, function is the inverse z-transform of a perfect lowpass filter. It cannot be used directly since it extends infinitely far into both forward and backward time. A finite duration impulse response is obtained by multiplying the “perfect" response by a finite-duration window function. The one discussed here uses Chebyshev polynomials as their basis. These polynomials are discussed in Appendix B They all have the property that the polynomials' peak magnitude is unity for values of x between -1 and 1, and that for greater values of $|x|$, the magnitude grows as x^M where M is the order of the polynomial. One such polynomial is shown in [link].



A Chebyshev Polynomial
(drawn from [1])

We desire that the oscillatory portion of the polynomial correspond to the stopband region of the filter response and the x^M portion to correspond to the transition from the stopband to the passband. This is accomplished by invoking a change of variables relating x to the frequency f. The resulting equation is then evaluated at the several points to obtain an expression for the transition bandwidth $\Delta\ f$. The details of this manipulation are contained in Appendix C. They result in the following equation:

Equation:

$$\Delta f = \frac{f_s}{\pi(N-1)}\left[cosh^{-1}\left(\frac{1+\delta_1}{\delta_2}\right) - \left\{\left(cosh^{-1}\left(\frac{1+\delta_1}{\delta_2}\right)\right)^2 - \left(cosh^{-1}\left(\frac{1-\delta_1}{\delta_2}\right)\right)^2\right\}^{\frac{1}{2}}\right].$$

If δ_1 is small compared to unity and N is large compared to unity, as already assumed, then Δf is closely approximated by
Equation:

$$\Delta f = \frac{f_s}{\pi N}\left(cosh^{-1}\left(\frac{1}{\delta_2}\right)\right).$$

When the argument of the hyperbolic cosine is large, the function can be approximated as
Equation:

$$\frac{1}{\delta_2} = cosh\ y \approx \frac{e^y}{2}$$

With suitable manipulation we find that
Equation:

$$y \approx log_e \frac{2}{\delta_2} = log_e 2 - log_e \delta_2.$$

Substituting this expression for the inverse hyperbolic cosine yields a simple formula for Δf:
Equation:

$$\Delta f = \frac{f_s}{\pi N}(log_e 2 - log_e \delta_2).$$

Rewriting this equation shows that N must equal or exceed:
Equation:

$$N \geq \frac{\alpha f_s}{\Delta f}$$

where α is given by
Equation:

$$\alpha = \frac{log_e 2 - log_e \delta_2}{\pi}.$$

Rewriting equation 4 from the module titled "Statement of the Optimal Linear Phase FIR Filter Design Problem", δ_2 can be written as
Equation:

$$\delta_2 = 10^{\frac{-SBR}{20}} = e^{\frac{-2.303 \cdot SBR}{20}}.$$

Substituting this into [link] yields

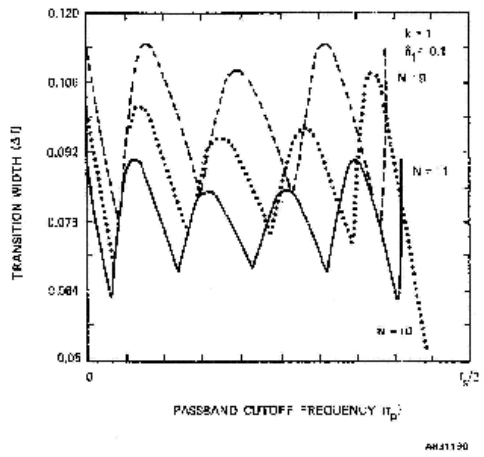
Equation:

$$\alpha = 0.22 + 0.0366 \cdot SBR,$$

which can be recognized as [link].

Caveats

The derivation just presented assumes that the filter of interest is a lowpass design, the filter order is high (> 20 or so), that the passband ripple is small (that $\delta_1 \ll 1$), and that the filter uses all degrees of freedom except one in the stopband, that is, that the filter has the lowest possible cutoff frequency. In fact not all of these conditions have to be met to make the design [link] and [link] useful. An indication of how errors can enter the estimate of N under other conditions can be seen, however, by examining [link].



Comparison of the Transition Widths of Even and Odd Optimal Lowpass Filters (drawn from [1])

This figure shows the smallest value of Δf attainable with optimal equal-ripple linear phase filters of different lengths as a function of the cutoff frequency f_c. [link] and [link] predict that the transition bandwidth is constant as a function of cutoff frequency and that it always gets smaller as the filter order N increases. [link] shows that these generalities are not true. It can be seen that Δf varies somewhat as a function of f_c and that there are particular choices of f_c where a lower value of Δf is actually attainable with a lower filter order rather than a higher one. It would appear that, for a given filter order N, some values of f_c are “hard" to attain a small transition bandwidth and others are “easy". This is in fact true and the reason for it will be discussed in "Why does alpha Depend on the Cutoff Frequency fc?".

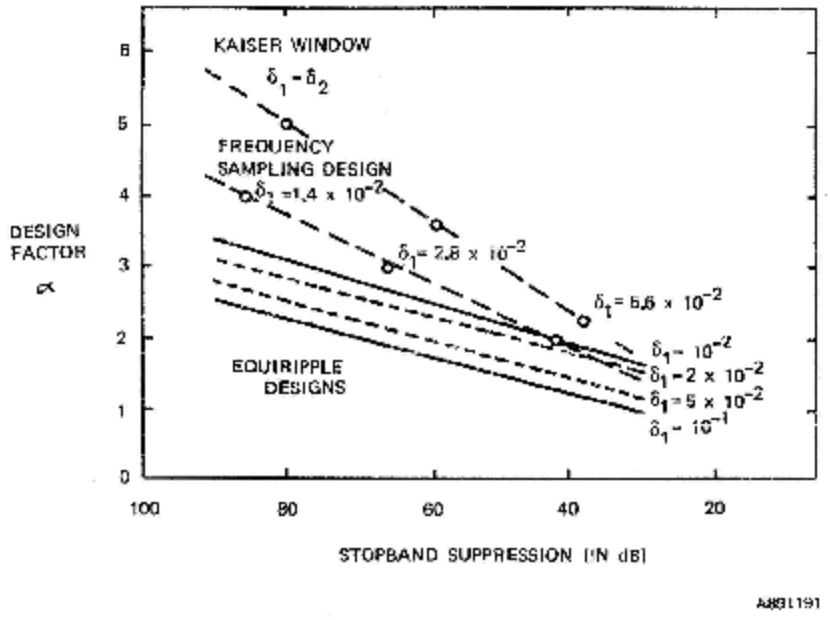
While [link] shows that Δf is not truly independent of the cutoff frequency f_c and monotonic in the filter order N, the significant variations appear only for low filter orders. If N is greater than 20 or so, and the other conditions listed above hold true, as they usually do, then [link] and [link] can be used with impugnity, even for highpass and bandpass filters.

Performance Comparsion with other FIR Design Methods

Performance Comparsion with other FIR Design Methods

A commonly asked question among filter designers is why should the optimal design methods be used at all, or, equivalently, how much does the use of an optimal technique buy over some other conventional methods. This question is conveniently answered using [link], a figure extracted from [1] and modified to use the definitions of variables employed in this technical note. The figure shows the value of the design parameter α needed to attain a specific degree of stopband suppression in lowpass filters. Since the filter order N and therefore the amount of computation[footnote] $R = Nf_s$ are directly proportional to α, it serves as an excellent indicator for comparisons.

The actual amount of computation depends on whether the data is real- or complex-valued, whether the impulse response symmetry is exploited, and whether interpolation or decimation is used. In all cases, however, R is proportional to f_s and α, and therefore [link] provides an accurate indication of the relative computational complexity of the filters resulting from the different design methods.



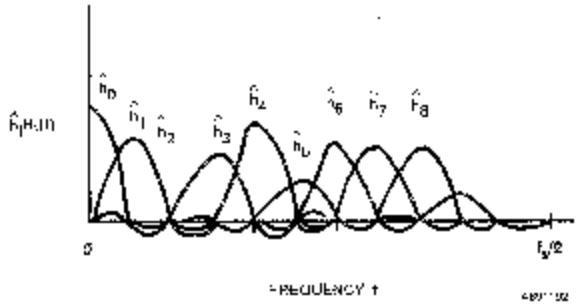
Comparisons among Windowed, Frequency Sampling, and Optimal Lowpass Filters (drawn from [1])

Curves for three design methods are shown, windowing techniques, so-called “frequency sampling" techniques, and the optimal, equal-ripple design produced by the Parks-McClellan program. In each case there are some variations depending on the choice of design parameters other than stopband ripple. For example, the optimal technique shows a band of results indexed by the amount of passband ripple (hence δ_1) specified. The figure shows that, for modest degrees of stopband suppression, all of the methods work about equally well. For high degrees of suppression, however, the optimal technique allows values of α to be attained which are on the order of half of those attainable with the windowing methods and about 60-70% of the frequency sampling method. Since computation is directly proportional to α, these saving are directly translatable into hardware and/or runtime improvements.

Why, one might ask, is the optimal method significantly better than, say, the window method? A fuller answer is presently shortly, but a simple one is that the optimal methods allow the designer to avoid overdesigning portions of the frequency response about which he or she needn't exert as much control. For example, recall the design example discussed in the section "Conversion of Specifications" from the module titled "Statement of the Optimal Linear Phase FIR Filter Design Problem". In that case a set of reasonable specifications was developed which allowed the magnitude of the passband ripple to be almost 29 times larger than the stopband ripple. Since the Parks-McClellan design package allows the design of **weighted** equal-ripple filters this disparity can be accommodated. Window-designed filters, however, are constrained to have exactly the same passband ripple δ_1 as stopband ripple δ_2. Effectively the optimal design methods allow the degrees of freedom in the impulse response to be focused on the most stressing parts of the frequency response design while the window method treats all parts equally. The frequency-sampling method falls in between.

The Meaning of the Design Parameter α

More insight into the meaning of the design parameter α can be gained by examining all three aforementioned design methods in terms of the inverse discrete Fourier transform. Suppose that our objective, as it is, is to synthesize an N-point FIR filter. Suppose further that we use the approach of specifying the frequency response we desire with equally spaced samples in the frequency domain and then use the inverse discrete Fourier transform (DFT) to transform the frequency specification into a time-domain impulse response. This approach is shown in graphical form in [link].



Using the Discrete Fourier Transform (DFT) as the Basis of FIR Filter Design

Analytically there is a one-to-one relationship between the N points of an FIR impulse response and the frequency response of the filter measured at N equally-spaced frequencies between 0 and f_s Hertz. Specifically it is straight-forward to show that the impulse response $h(k)$ and the complex gains $\widehat{h}_n$, for $0 \leq n \leq N-1$, are invertibly related, where the filter's frequency response is given by

Equation:

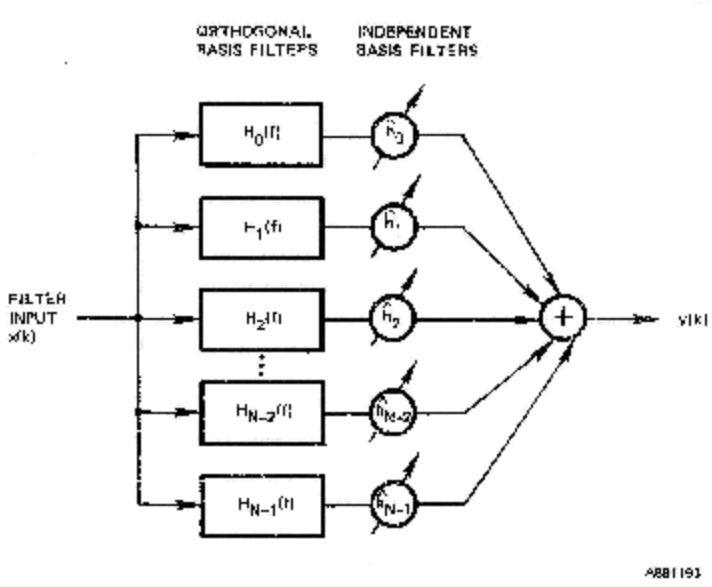
$$H(f) = \frac{1}{N} \sum_{n=0}^{N-1} \widehat{h}_n \frac{sin\ \pi(NfT - n)}{sin\ \pi\left(fT - \frac{n}{N}\right)}.$$

Thus choosing the complex gains $\widehat{h}_n$ is equivalent to choosing the impulse response $h(k), 0 \leq k \leq N-1$, and, through [link], to the filter frequency response at all values of f between 0 and f_s Hertz. By examining [link] it can be seen that choosing a frequency response (and hence an impulse response) can be intuitively viewed as adjusting the gain levers on a graphic equalizer of the type now used on home stereos. Each lever sets the gain, denoted here as $\widehat{h}_n$, of a filter given by
Equation:

$$H_n(f) = \frac{1}{N} \frac{sin\ \pi(NfT - n)}{sin\ \pi\left(fT - \frac{n}{N}\right)}.$$

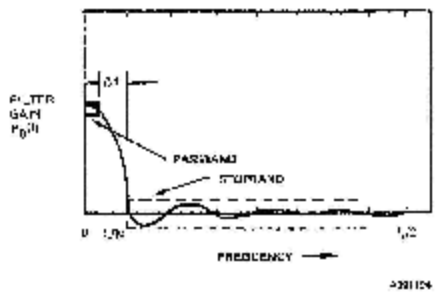
By setting these N gain values optimally the best possible frequency response is attained.

The analogy of the graphic equalizer can be followed somewhat further. [link] suggests that the FIR design problem can be thought in the terms of the structure shown in [link]. The input signal is applied to all N of what we'll the **basis** filters, where the frequency response of the n-th filter is given by [link]. As noted earlier these basis filters, so called because they form the linearly independent set of filters used to construct $H(f)$, are frequency-shifted versions of the same fairly sloppy bandpass filter. These filter outputs are then scaled by the complex coefficients $\widehat{h}_n$ and then added together to produce the observable filter output. Thus the basis filters are fixed and the $\widehat{h}_n$ control the frequency and hence impulse response of the digital filter. It should be noted that the filter is not usually actually constructed[footnote] as shown in [link] but it is a very convenient analogy when trying to understand the relationships between the various filter synthesis methods.
Frequency-domain filters are of course the counterexample.



The FIR Filter Design Problem Models as a Bank of Bandpass Filters

Now we shall use the model. In our quest for the true meaning of α, consider first the design of a simple lowpass filter. We desire the cutoff frequency f_c and the stopband edge f_{st} to be as low as possible and allow the peak stopband ripple to be quite large. Using the graphic equalizer model just discussed yields the design shown in [link]. Only one filter, the one centered at DC, is used. Its gain is set to unity and that of all others is set to zero. The peak stopband ripple is determined by the first sidelobe of the only active filter. It can be computed to be about 13 dB below the maximum passband power level (measured at DC).



A Simple Lowpass Filter Designed Using the Graphic Equalizer Analogy

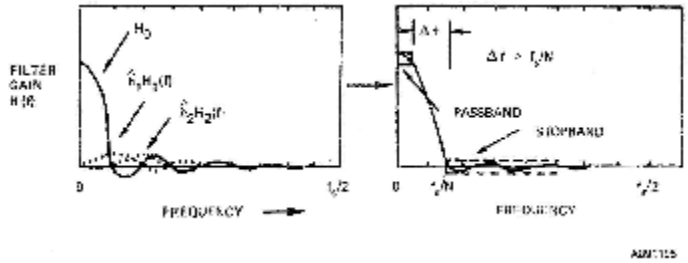
What is Δf in this case? Graphically it can be seen to be somewhat less than than the frequency interval between DC and the first transmission zero of $H_n\,(f)$ which occurs at $f = \frac{f_s}{N}$. Suppose that we now rewrite equation 2 from the module titled "Filter Sizing" as

Equation:

$$\Delta f \;\approx\; \alpha \frac{f_s}{N}.$$

Thus we see that in the simple filter designed in [link] that associated value of α is slightly less than one.

Now suppose that we attempt to design a better filter, again using the graphic equalizer method. Our first objective is to reduce the size of the stopband ripple. To do this we leave $\widehat{h}_0$ set to unity and increase the values of $\widehat{h}_1$ and $\widehat{h}_2$ slightly so that their positive mainlobe values cancel the negative-going first sidelobe of $\widehat{h}_0$. All other filter gain levels will remain set to zero. The effects of this strategy are seen in [link].



Lowpass Filter Obtained Using the Second and Third DFT Basis Functions

The first objective, that of reducing the peak stopband ripple, is achieved. By choosing $\widehat{h}_1$ and $\widehat{h}_2$ just right, the first sidelobe of $\widehat{h}_0$ can be effectively cancelled, leaving the other sidelobes to compete for the peak value. The second effect is less desirable, however. From graphical inspection it is clear that Δf, the frequency interval between f_c and f_{st}, has grown. It now exceeds $\frac{f_s}{N}$, thus making α greater than unity.

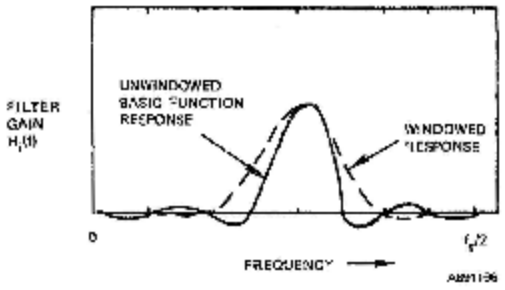
These trends continue as more and more filter gains $\widehat{h}_n$ are allowed to become non-zero in the quest of further reducing the peak stopband ripple. The peak is reduced, the ripple structure begins to approach the Chebyshev equal-ripple firm seen in Figure 1 from the module titled "Statement of the Optimal Linear Phase FIR Filter Design Problem", and the transition band stretches out as more filters are used to try to constrain the stopband frequency response to the stopband ripple goals. The design parameter α is just a measure of the number of filters, or, equivalently, the number of equalizer levers, needed to transit from one gain level (e.g., the passband) to another (e.g., the stopband) while achieving the desired passband and stopband ripple performance. Since $\frac{f_s}{N}$ is the spacing between the bins of an N-point DFT, the term α can also be thought of as the number of DFT bins needed to make a gain transition. This interpretation is explored next.

Three Methods of Designing FIR Filters

The module "Performance Comparison with other FIR Design Methods" alluded to the fact that three basic methods have traditionally been used for the design of FIR digital filters. Figure 1 in the module titled "Performance Comparison with other FIR Design Methods" in fact compares their relative performance in terms of the value of α (which was shown to be proportional to the filter's required run-time computation rate). Given the background of the previous subsection it is now possible to understand each of the methods and to gain some insight into the differences between their performance.

Window-based Filters

As described earlier, one of the first class of FIR filters is that based on the use of a “smoothing window". This window, constructed to have only N non-zero points, is multiplied point-by-point by an impulse response of infinite duration which has the “perfect" frequency response. This multiplication or **windowing** has the effect of making the filter impulse response finite in duration (hence FIR), but also has the effect of smearing the desired frequency response.



The Effect of a Window Function on the Basis Filter

The stopband ripple specification is obtained by using a window capable of suppressing all sidelobes to the desired degree. This can be seen in [link]. The windowed filter basis function has substantially lower sidelobes than the original $\frac{sin\ Nq}{sin\ q}$ filter basis function, in trade for substantial widening of the main lobe. This widening means growth in the equivalent design parameter α and is monotonic with the degree of sidelobe suppression attained.

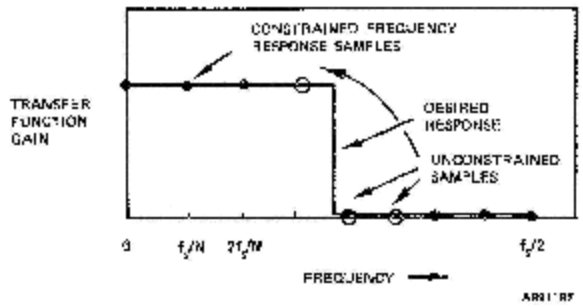
It should also be observed that the sidelobe reduction has the effect of reducing the ripple in the passband as well as in the stopband. Thus some of the filter's degrees of freedom are given up in perhaps overdesigning the passband response rather than focusing them on the stopband performance.

Frequency Sampling Design

In the simplest DFT-based FIR filter design method, the desired frequency response is sampled at frequency intervals of $\frac{f_s}{N}$ Hertz and the filter gains $\widehat{h}_n$ are set to those values. This is in essence the method used for the simple lowpass filter shown in Figure 4 from the module titled"Performance Comparison with other FIR Design Methods". The big advantages of this method are its simplicity and the fact that any desired response, no matter how complicated, can be approximated. The big disadvantage is its uncontrolled ripple performance in both the stopband and passband. The traditional cures for this are the use of a window function to suppress the ripple and the expansion of the filter order N to compensate for the window's smearing of the desired response. Increasing N, of course, increases the filter's run-time computation rate.

Relatively early in the development of FIR design techniques it was discovered that much better adherence to the desired frequency response could be attained by allowing some of the basis filter gains $\widehat{h}_n$ to vary slightly from the exact sampled values (e.g., 1 and 0 for a lowpass filter). This idea is shown in [link]. A simple lowpass filter is the desired response. Solid dots show the frequency samples of this desired response taken every $\frac{f_s}{N}$ Hertz. These samples have values of 1 and 0 for $\widehat{h}_n$ in the passband and

stopband respectively. Now suppose that the values of $\widehat{h}_n$ for n in the vicinity of the cutoff frequency f_c are allowed to be modified slightly with the goal of minimizing the peak stopband ripple. These values of n are denoted with small circles instead of solid dots in [link]. Rabiner and his coworkers [4] showed in 1970 that it was possible to use the linear programming optimization technique to manipulate two or three of the filter gains to obtain great improvement in stopband ripple performance. The computational complexity of the linear programming method, however, limited the number of the $\widehat{h}_n$ which could be so chosen.



Comparison of “Frequency Sampling" and Equal-ripple Design

Equiripple Design

It was generally known in 1971 that equal-ripple passband and stopband behavior would lead to the best filter performance, where “best" means the smallest transition band (and hence α) for a given set of peak passband and stopband ripple specifications. In fact a great deal was known about the properties of such filters. What was lacking was a computationally satisfactory method of designing such optimal filters. As just noted, the linear programming technique provided a big step but still fell short. The breakthrough came in two parts. Several workers, but principally Parks, McClellan (Parks' graduate student), and Rabiner showed that four different

variants of FIR linear phase filters could all be represented by the same set of equations[footnote] and could therefore be solved the same way. The second part was Parks' suggestion of using the the Remez exchange algorithm for doing the actual optimization. The Remez exchange algorithm effectively allows all degrees of freedom in the filter impulse response to be adjusted simultaneously while the linear programming technique allows the adjustment of only one at a time. For high order filters this distinction makes a tremendous difference in the number of computations needed to iteratively optimize a design. Refering again to [link], the Remez algorithm allows all of the frequency samples to be modified, even for filter orders as high as $N = 1000$ or more, thus permitting the best possible filter performance to be achieved. McClellan also proved that the linear phase FIR filter design problem satisfied the conditions needed to guarantee convergence of the Remez algorithm.

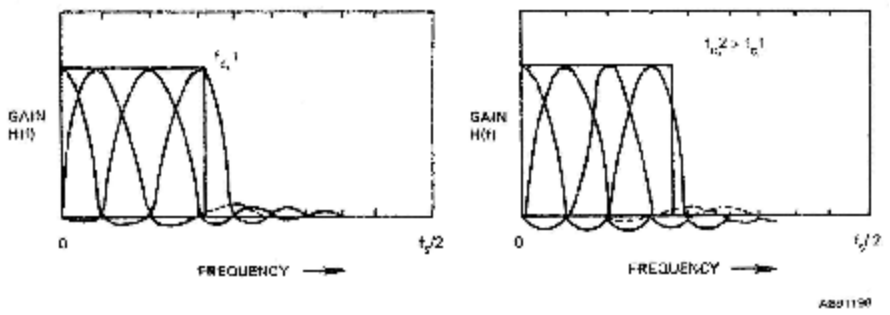
The variants are odd and even filter order and symmetric and antisymmetric impulse responses.

Why Does α Depend on the Cutoff Frequency?

The formulas presented in Equation 1 and Equation 2 from the module titled "Filter Sizing" imply that α and hence the required filter order N are independent of the cutoff frequency f_c. The supporting analysis showed that this is only true in the limit of high order filters, i.e. when N is large. The dependence for shorter filters is shown in Figure 2 from the module titled "Filter Sizing". Why should this occur? Consider the filter design problem shown in [link]. Again the goal is a simple lowpass filter with cutoff frequency f_c. The frequency sampling points at frequency multiples of $\frac{f_s}{N}$ are also shown as solid dots. Instead of fixing the gains we presume that the filter gains $\widehat{h}_n$, or, equivalently, the graphic equalizer levers, are optimized, by whatever means, to yield the best stopband ripple performance.

[link](a) shows the combination of gains $\widehat{h}_n$ needed to constrain the peak stopband ripple to a given level, say δ_2. The frequency at which this equal ripple band starts is of course f_{st} and the difference between f_{st} and f_c is Δf. Now suppose that f_c is increased slightly, as shown in [link](b). Now a different set of the $\widehat{h}_n$ are needed to make the peak ripple equal δ_2 and these result in different values of f_{st} and Δf. Pursuing this graphical analysis we find that:

- Cutoff frequencies near multiples of $\frac{f_s}{N}$ result in smaller transition bands, and hence smaller values of α, than those near the center of two bins. This occurs, to first order, since two or more stopband basis filters are needed to cancel the first sidelobe of the last basis passband filter when the passband stops between two bins, while one is needed if the passband stops near a bin.
- Because these “hard" and “easy" frequency ranges occur for every bin, the number of the ranges, counting both positive and negative frequencies, is about the same as the filter order[footnote]N. Various boundary conditions can make the actual number one less or one more than the filter order.
- The variation in the transition band Δf is more pronounced as N decreases since there are fewer basis filters to use in optimizing the response.



Visualizing the Effects of Cutoff Frequency on Design Difficulty

As an aside one might observe from Figure 1 from the module titled Performance Comparison with other FIR Design Methods that all three methods perform about equally for high levels of stopband ripple. Intuitively the reason for this should now be clear. Window-based methods need not use much shaping if high levels of ripple are tolerable. Similarly, frequency sampling need not use many adjustable coefficients. Since this is true the equal-ripple techniques will not perform much better since their only advantage is that of adjusting all of the filter gains. The underlying point is that, for high-ripple designs, all of the methods produce designs closely resembling the sum of simple, shifted $\frac{sin\ Nq}{sin\ q}$ functions and produce a transition band Δf of about the order of $\frac{f_s}{N}$, hence an α of about unity. Only as the stopband ripple specification grows tighter does the method and accuracy of adjusting the coefficients and the number of them available for adjustment begin to affect the transition band performance.

Extension to Non-lowpass Filters

All of the discussion to this point has focused on lowpass filters. Practical applications require other types, of course, including highpass, bandpass, and bandstop designs. In fact the analysis presented in the previous sections applies to all of these design criteria and the rules for filter length estimation can be used almost directly. In general Equation 1 and Equation 2 from the module titled "Filter Sizing" apply when one of the equal ripple specifications dominates all others and when one of the transition band specifications dominates all others. As a practical matter this means that δ_i dominates if it is less than one-tenth of all other rippple specifications and that Δf_i dominates if it is simply less than all others. Suppose we define δ and Δf by the equations:

$$\begin{aligned} \delta &= min\{\delta_i\}, \text{for all pass and stopbands i, and} \\ \Delta f &= min\{\Delta \text{f}_k\} \text{for all transition bands k} \end{aligned}$$

Equation:

$$\delta = \min\{\delta_i\}, \text{for all pass and stopbands i, and}$$

Equation:

$$\Delta f = \min\{\Delta f_k\} \text{for all transition bands k}$$

If so then equation Equation 1 from the module titled "Filter Sizing" can be used directly and the equation for α becomes

Equation:

$$\alpha = 0.22 - \frac{log_e \delta}{\pi}.$$

A final hint - Watch out for the implicit boundary conditions present in the design of linear phase FIR digital filters in two cases: even order, symmetric response and odd order, antisymmetrical response. In both of these cases the underlying equations for the filter's frequency response constrain it to equal exactly zero at $\frac{f_s}{2}$. This is obviously not a problem for lowpass filters, since

the desired gain at $\frac{f_s}{2}$ is zero already. However, in the design of multiband and highpass filters an inordinate amount of engineering time has been spent trying to design even-order filters when in fact it is impossible to do so. The Parks-McClellan algorithm will gamely try, but will fail. As a rule, use odd values of N for highpass and multiband filters requiring nonzero response at $\frac{f_s}{2}$ and use even-order filters for differentiators.

Bibliography for "Notes on the Design of Optimal FIR Filters"

References

"Notes on the Design of Optimal FIR Filters" Appendix A

The Formula for Converting between and Passband Ripple

From equation 2 in the module titled Statement of Optimal Linear Phase FIR Filter Design Problem, the peak-to-peak passband ripple, measured in decibels, is given by
Equation:

$$PBR = 10\ log_{10}\frac{(1+\delta_1)^2}{(1-\delta_1)^2},$$

where δ_1 is the peak amplitude deviation in the passband. Suppose now that
Equation:

$$0\ <\ \delta_1\ \ll\ 1.$$

If so, then the passband ripple PBR is closely approximated by
Equation:

$$PBR \approx 10\ log_{10}(1+4\delta_1).$$

Now recall that $log_e\ (1+x)\ \approx\ x$, when x is small compared to unity, and that $log_{10}x\ \approx\ 0.434 \cdot log_e x$. Combining these facts, leads to the equation
Equation:

$$PBR \approx 10\ log_{10}(1+4\delta_1)\ \approx\ 4.34 \cdot log_e(1+4\delta_1)\ \approx\ 17.36 \cdot \delta_1.$$

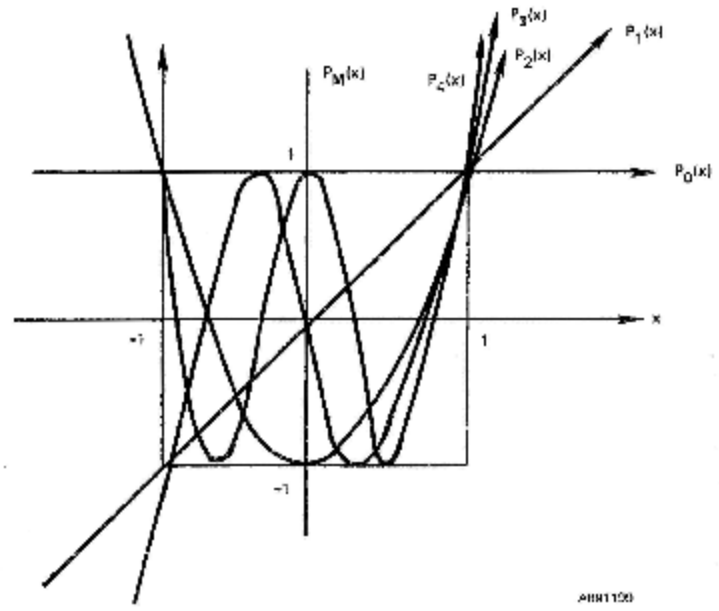
This formula holds as long as δ_1 is small compared to unity. Using $\delta_1 = 0.1$ as a benchmark, the formula holds for values of passband ripple less than 1.5 to 2 dB, the range in which most filter design falls.

"Notes on the Design of Optimal FIR Filters" Appendix B

Some Notes on Chebyshev Polynomials

The section "The Derivation of the Formula" from the module titled "Filter Sizing" used some of the properties of the Chebyshev polynomials to develop the key formulas used for FIR filter sizing. This appendix provides a very brief review of these polynomials and the equations used to generate them.

[link] shows a set of polynomials which have the property that, for values of x between -1 and 1, the polynomial has peak magnitude of unity. A footnote in The section "The Derivation of the Formula" from the module titled "Filter Sizing" pointed out that the Russian engineer Chebyshev developed these polynomials as part of design effort which required minimizing the maximum lateral excursion of a locomotive drive rod. For each polynomial order, say M, the objective is to choose the polynomial's coefficients so that that it “ripples" between $x = -1$ and $x = 1$ and then proceeds off proportional to $|x|^M$ for values of $|x| > 1$. Not only did Chebyshev find such polynomials, he found that one exists for each positive value of M, and that they are related thorugh a recursion equation, that is, the polynomial for M can directly obtained for the polynomial for M-1.



Graphs of Chebyshev Polynomials of
Orders 0 through 4

Consider the following recursion expression:
Equation:

$$P_M(x) = 2x \cdot P_{M-1}(x) - P_{M-2}(x),$$

with initial conditions of
Equation:

$$P_0 = 1$$

and
Equation:

$$P_1 = x$$

Note that both of these initial conditions meet (if trivially) the stated criteria for being Chebyshev polynomials.

Using this recursion expression we find, for M from 0 to 5, that:
Equation:

$$\begin{aligned} P_0(x) &= 1 \\ P_1(x) &= x \\ P_2(x) &= 2x^2 - 1 \\ P_3(x) &= 4x^3 - 3x \\ P_4(x) &= 8x^4 - 8x^2 + 1 \\ P_5(x) &= 16x^5 - 20x^3 + 5x \end{aligned}$$

These polynomials are plotted in [link] and it may be confirmed by inspection that they meet the stated criteria.

A surprising result is that there is yet another way to present these polynomials. This method is given by the following equations:
Equation:

$$P_M(x) = cos\left[M \cdot cos^{-1}(x)\right], \ for \ \left|x\right| \leq 1, \ and$$

Equation:

$$P_M(x) = cosh\left[M \cdot cosh^{-1}(x)\right], \ for \ \left|x\right| > 1.$$

Analytically it can be confirmed that these equations satisfy the recursion seen in equation [link]. To see that they describe the same polynomials as seen in [link], consider [link] for values of $\left|x\right|$ between -1 and 1. For such values $cos^{-1}x$ ranges between π and 0. Thus $M \cdot cos^{-1}x$ ranges between $M\pi$ and 0, and $cos\left[M \cdot cos^{-1}x\right]$ cycles between -1 or 1 and 1, hiting $M + 1$ extrema on the way, counting the endpoints. Similar analysis shows that equation [link] grows monotonically in magnitude as $\left|x\right|$ does. In fact it

is easy to show that $\left|cosh\left[M \cdot cosh^{-1}x\right]\right|$ assymptotically approaches $|x|^M$ as $|x|$ gets much greater than one.

This second form of the definition for Chebyshev polynomials is very useful since it is a closed form and because it involves cosines, a functional form appearing frequently in frequency-domain representations of filters. In light of this a final twist might be noted. [link] is in fact superfluous given [link]. To see this, consider evaluating [link] for $|x| = 2$. It initially appears that this won't work, since arccosine cannot be evaluated for arguments greater than unity. In fact it can, it's just that the result is purely imaginary. It is easy, using Euler's definition of the cosine, to see that the cosine of jx is the same as the hyberbolic cosine of x. Thus the arccosine of 2 is j times the inverse hyperbolic cosine of 2, that is, $j \cdot 1.31$. Multiplying by M and taking the cosine of the product yields the cosine of jMx, which is the hyperbolic cosine of Mx. Thus, if imaginary arguments are permitted, then [link] suffices to describe all of the Chebyshev polynomials.

"Notes on the Design of Optimal FIR Filters" Appendix C

Using a Chebyshev Polynomial to Estimate

We desire that the oscillatory portion of the polynomial shown in Figure 1 in the module titled "Filter Sizing" correspond to the stopband region of the filter response and the x^M portion to correspond to the transition from the stopband to the passband. This is achieved by employing a change of variables from frequency f to the polynomial argument x:
Equation:

$$x \quad \frac{x}{\quad} \quad cos \quad \frac{\pi f}{f_s} \quad \frac{x}{\quad}$$

While many different types of variable changes could be employed, this one matches the boundary conditions (an obvious requirement) but happens to employ the cosine function, a member of the same family used to define the Chebyshev polynomials.

With this change of variables we see that the transition band Δf is defined by the difference between x and $x \quad x_p$. Using the closed, but nonintuitive form of the K-th order Chebyshev polynomial, valid for x , we have that
Equation:

$$P_K \; x \quad cosh \; K \; cosh \quad x$$

To synthesize the desired impulse response using this windowing technique we multiply the resulting window function by the sampled sinc function. In this case, however, we desire that the cutoff frequency be as low as possible, limiting at zero Hz. The associated sinc function equals unity for all non-zero coefficients of the impulse response. Since the final impulse response is the point-by-point product of the window and the sampled sinc function, in this case the window itself is the resulting impulse response. It suffices then to examine the properties of the N-th order Chebyshev polynomial to see how the N-point optimal filter will behave.

To find the relationship between the required filter order N and the attainable transition band Δf, we first determine the proper value of K and then evaluate [link] at the known combinations of x and $P_K \; x$. To select K we note that all but one of the ripples in the polynomial's response are used in the stopband and these are split evenly between the positive and negative frequencies. Thus a filter and window of order N implies a Chebyshev polynomial of order
Equation:

$$K \quad \frac{N}{\quad}$$

With this resolved we observe from Figure 1 in the module titled "Filter Sizing" that
Equation:

$$P_{N\_}$$

Equation:

$$P_N \quad x_p \qquad \frac{\delta}{\delta}$$

Equation:

$$P_N \quad x \qquad \frac{\delta}{\delta}$$

These equations are manipulated to yield an expression for x_p. [link] is then used to obtain values for f_{st}, corresponding to x , and f_c, corresponding to $x \quad x_p$. Their difference, defined earlier to be the transition band Δf, is then given by

Equation:

$$\Delta f \quad \frac{f_s}{\pi \ N} \quad cosh \quad \frac{\delta}{\delta} \qquad cosh \quad \frac{\delta}{\delta} \qquad cosh \quad \frac{\delta}{\delta}$$

Under suitable conditions this equation can be simplified considerably. For example, in the limits of small δ and large N, [link] reduces to Equation 4 in the module titled "Filter Sizing".